MATLAB PROJECT 2

1a.

```
n = 10;
x = zeros(1,n);
sum = 0;
for i = 1 : n;
    x(i) = i^2;
    sum = sum + x(i);
end
sum
```

```
>> loop1

sum =

  385
```

Line 4: Sets up a for loop that will iterate from ‘I = 1’ to ‘I = n’. Loop increments 1 value after it runs

Line 5: X is being set equal to the current value of I squared and storing it in the previously made array

Line 6: Adds number from previous line to the current value of sum

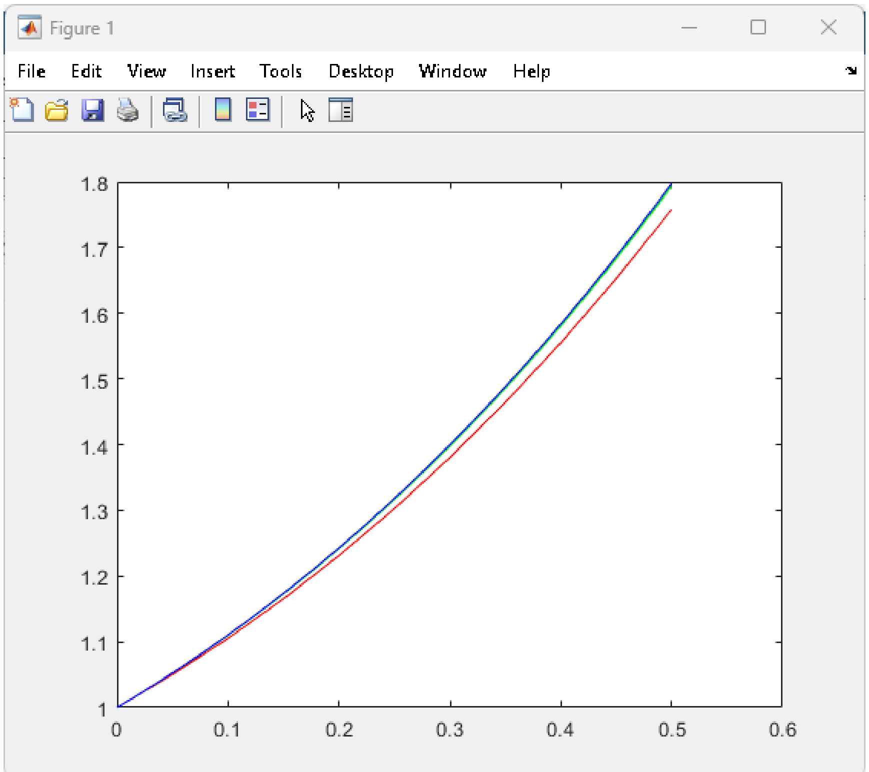
Line 7: Marks end of for loop, repeats the same loop until the last value is used (n)

1b.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% MATH 238 Project2 Q1 %%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

close all;
clear all;

%%% Define the interval [a,b] and initial condition
a = 0;
b = 0.5;
y0 = 1;

%%% Find the approximated solution using Euler's method
N = 10;                                                 %%% number of steps
[x1,y1] = Euler(a,b,y0,N);                              %%% use the function
"Euler" to find the approximate solution with step size h=0.05

N = 100;                                                %%% number of steps
[x2,y2] = Euler(a,b,y0,N);                              %%% use the function
"Euler" to find the approximate solution with step size h=0.005

%%% Define the exact solution
N = 100;
x3 = a:(b-a)/N:b;
y3 = 2.*exp(x3) - x3 - 1;
```

```
%%% Plot the 3 solutions on the same figure
figure;
plot(x1,y1,'r'); hold on;
plot(x2,y2,'g'); hold on;
plot(x3,y3,'b'); hold off;


%%% Euler's method
function [x,y] = Euler(a,b,y0,N)

f = @(x,y) x+y;

h = (b-a)/N;

x = a:h:b;
y = zeros(1,N+1);

x(1) = a;
y(1) = y0;

%%% main loop for Euler's method
for i = 2:N+1
    x(i) = x(i-1) + h;
    y(i) = y(i-1) + h*f(x(i-1),y(:,i-1));
    %%% enter commands here so that the loop computes the approximate solutions %%%
end

end
```

```
>> Project2_Q1
```



2a.

Dy/dx = 2x+y

(Dy/dx) – y = 2x

Mu = e^-x

(d/dx)(e^-xy) = int(2xe^-xdx)

Y = -2x-2+Ce^x

-2 = -2(0+1)+Ce^0

C = 0

Y = -2x-2

2b.

3 = -2(-1)-2 + Ce^-1

C = 3e

Y = -2x-2+3e^(1+x)

2c.

0 = -2(-2)-2+Ce^-2

C = -2e

Y = -2x-2-2e^(1+x)

2d.

As x goes towards positive infinity, y will go to positive infinity as well at an increasing exponential rate.

As x goes towards negative infinity, y will decrease.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% MATH 238 Project2 Q2 %%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear all;

%%% Define f(x,y) in the IVP
f = @(x,y) 2*x+y;

%%% Draw the direction field
figure;
dirfield(f, -3:.3:3, -4:.4:4);hold on;

%%% Define the solutions to the IVP in Q2(a)-(c)
x = -3:0.1:3;
y1 = -2*(x+1);
y2 = -2*(x+1) + 3*exp(1+x);
y3 = -2*(x+1) - 2*exp(2+x);

%%% Plot the 3 solutions on the direction field
plot(x,y1,'r');hold on;
plot(x,y2,'g');hold on;
plot(x,y3,'b');hold off;
```

```
>> Project2_Q2
```

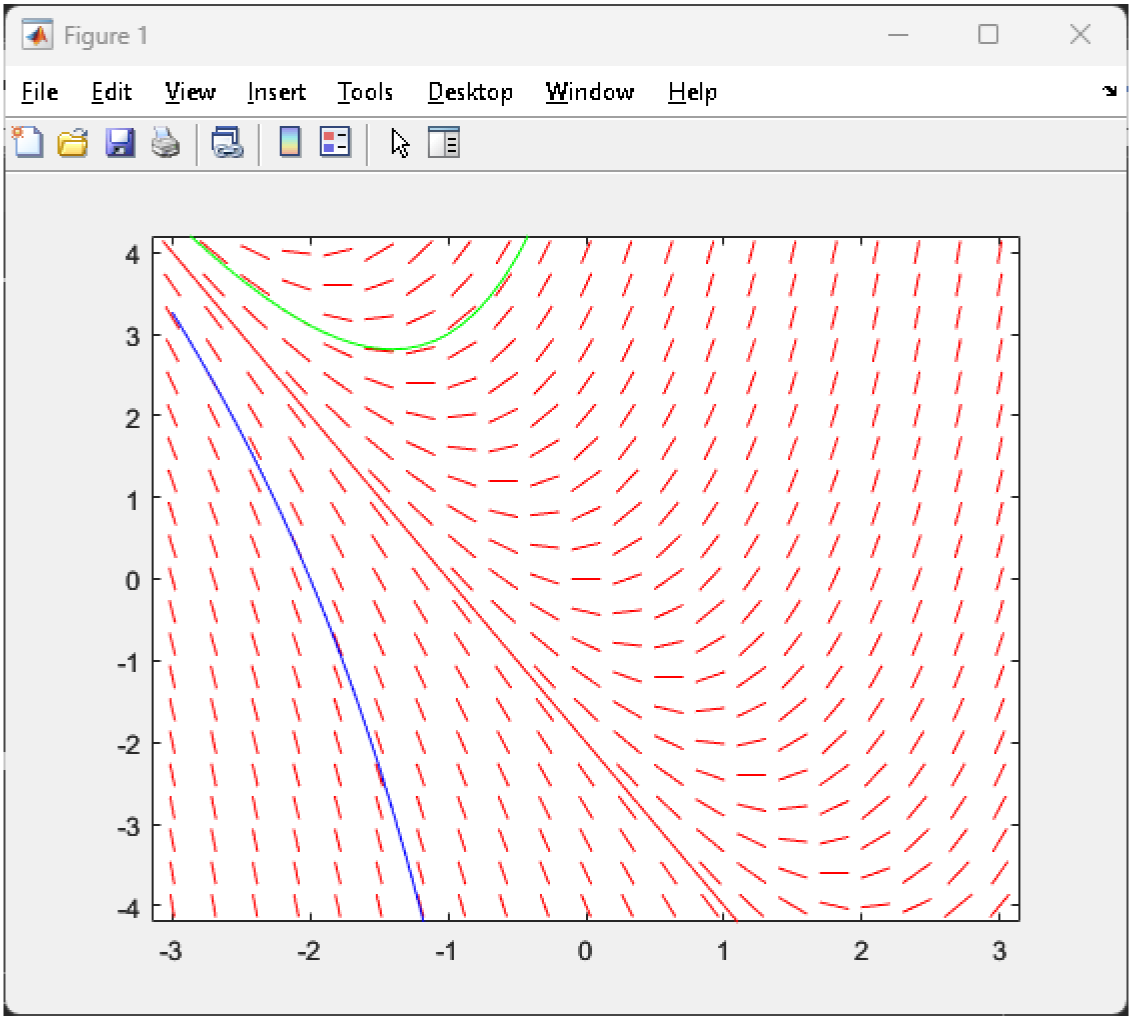


Figure 1
File
Edit
View
Insert
Tools
Desktop
Window
Help
4
3
2
1
0
-1
-2
-3
-4
-3
-2
-1
0
1
2
3